ISSN 0112-3351

BEEBLET

CONTENTS:

Magazine Hijack 2
Classified Ads 2
President Report - Modems 3
- Master 4
Easter 7
ADFS III 8
Machine code print 11
Something Bigger II 12
Electron Column 16
Fault finding 17
Pressing problem 19
Electron Viewsheet Index 20
Modem Owners 22
Acorn Speech 23
Elsewhere in the House 24
Snowball 26
Adventure Games - PAS 27
Z80 co-processor 32
Letterbox 34
Program scrambler 38
Wellington Group news 39

VOLUME 4 NUMBER 4 MAY 1986

BBC / ACORN

COMPUTER USER GROUP NZ (inc)

MAGAZINE HIJACKED!!

Monday 5 May: At work, Bruce's fiancee is ordered to get him to make a phone call. That evening:

"Give me everything you've got for the May magazine if you want a May mag for the printers by next Monday" he is commanded.

"But I've only got about 2 pages" he complains.

"Plus the dozen or so I've typed up here for you" he is told.

OK. The insane idiot on the other end of the phone spends the next 2^ hours on the phone forcing deadlines on people who've been meaning to write something but haven't got around to it. "Thursday at the local meeting! Be there! Or else!"

Tuesday 6 May: Bruce brings discs, untyped articles round.

Wednesday & Thursday were quiet. Idiot makes son a suit.

Thursday evening, Royal Society rooms: People have been very cooperative, and material pours in... or else: "You'll have to dictate it over the phone Saturday morning!!" they're told.

Monday 12 May 7.30am-ish Bruce takes magazine to printers.

This month's "Editor" and Hijacker is Anne Flower. I was the first editor of Beeblet back in the days of PRINT statements, cassettes and 0.1 OS bugs. Discs and Wordwise+ are quicker.

VERY LARGE Thank Yous to all who got their arms twisted, and responded with an abundance of material. To those of you who live a toll call away, sorry I didn't twist your ears - I'm twisting them now:

DEADLINE for each month has always been the last day of the previous month, so Queens Birthday weekend look out! It may see a June magazine being produced. If it doesn't? I'll have had an NMI more than a week early (baby). Bruce, getting married 24 May, isn't due back from honeymoon until 10 June so I've offered to at least start the June mag if YOU all help.

CLASSIFIED ADS:

BBC B with DFS, Wordwise, View + tape recorder $900ono phone Wgtn 759-958

BBC B with DFS, Wordwise, Speech $800ono
3^" drive 40tk $250ono
12" green monitor $125ono
Ph. Neville Hudson Wgtn 882-444

Offers wanted for:
1. Solidisc 128 SWR
2. BBC B, solidisc DDFS 2.0, 128 SWR, View, Viewsheet
3. BBC B, Solidisc DDFS 2.0, 128 SWR, View
Contact David Wgtn 759-006 evg or wKend

PRESIDENT'S REPORT - MAY 1986

MODEMS

At long last, all the modems have been made and sold. Had I known what I was getting in for when I started this project a year ago I probably wouldn't have bothered. My thanks to both Tim Lawson and Tim Ryan for all the work they did on the project.
To avoid sales tax complications we made only 50 modems, and are unlikely to make any more. The first 25 were delivered just before the end of the financial year, and the second 25 were sent off in the middle of April. There was a small wiring fault in the first 25 - all the owners have been advised how to fix it. (In Auckland, Gary at 4446694 business or 4447907 home has offered to help anyone who wants to make the modification.) Also, all modems have a minor problem in that CTS will not come on when the modem is in 300/300 ORIGinate mode unless the 1200/75 switch (which shouldn't be relevant) is left up.

A more major problem is that the modem appears to be affected by radio frequency interference in areas near powerful radio transmitters. The three modems near the Titahi Bay transmitters seem to interpret the RF as carrier and produce random input characters. We are working on that problem, and would like to hear from anyone else who has encountered it and has perhaps overcome it. I have a very strong interest in solving the problem - my modem is one of those that doesn't work!

Otherwise, the modems seem to work well. I have had a number of enquiries about what to use them for. I had envisaged a major use being to call up hobbyist Bulletin Boards. However, in Wellington at least there is currently none available to us. There are moves afoot to set one up, and with half of the modems having gone to Wellington members we may form a sizeable pressure group in helping to set one up. In Auckland, Steve Williams runs a bulletin board on 604290. There is also a NZ Micro club board on Auckland 397748, although Gary reports that there seem to be a lot of bugs in their board software. Please let us know the numbers of any other bulletin boards around the country and we will publish them here for you.

The other major use I envisaged is for communicating with other BBC people to share programs, perhaps run two-player games that require two computers to talk to each other (how come nobody has written anything like that yet - a good topic for a program competition?) send submissions to the Editor, and so on. Of course you will need to know who else has a modem. Elsewhere in this Beeblet is a list of first names and phone numbers of most of those who bought modems, so you will at least be able to get in touch with someone relatively local.

If you have the money, you could also subscribe to one of the Videotex services. ADITEL charges $50 to register, then $10 per month plus a usage time of 10c per minute. The Post Office will charge you an additional 10c per minute or so. COMPUTEX is a Closed User Group on Aditel, which is sponsored by Bits and Bytes - contact them for further information. The Post Office has also started up its own Videotex system and is starting to operate electronic mail services - ask them about it. You will also need Videotex software (e.g., COMMSTAR) - the free software provided with the VolksModem does not handle Videotex.

I AM NOT THE DESTINY OF MY OWN MASTER

It's rather a long story. Back in February I learned the specifications of the new Master just before departing on a business trip that took me through Britain. I had been planning to upgrade my BBC B with a sideways RAM board and perhaps a shadow memory board. After some quick calculations I decided that it would not cost a lot more to sell my BBC and upgrade to a Master while in Britain. I was fortunate to find a buyer for the BBC within a day. Having heard that there might be a shortage in Britain I 'phoned Technomatic to say I would be there in a week and could they PLEASE set a Master 128 aside for me. Sure - no problem - they said.

It was a different story when I arrived. Technomatic insisted that I had asked for a BBC B+128 and had set one of those aside. They refused to let me have a Master. So, I tried various other dealers. They all had waiting lists of 2 weeks or so and were getting very few Masters in. The story seemed to be that Acorn was selling most of the production to schools as they wanted to spend their budgets before the end of the financial year. Watfords would have sold me one, but would not take a VISA card (complained about the commission they had to pay) and I had no way of obtaining cash at the time. I finally located a possibility at a nearby dealer called P&H Electronics (they advertise in Micro User). They had one which a customer was late in picking up, and if he didn't come by lunchtime they said I could have it. I phoned back at lunchtime - it was still there and they said to come and get it. But by the time I got there he had turned up and they had given it to him.

I began to wonder at this time whether someone up there didn't want me to get another BBC, and came very close to buying an Atari 520 ST instead, which would have been a similar cost to a BBC with disk drives. However, I persevered and ordered a

Master and dual drives from P&H, in the hope that they would give me good service to make up for having sold the one they promised me. I must say that they did give me a good price on both the computer and the drives. Two weeks later, after I had arrived back in New Zealand they telephoned to check my VISA number and to say that they were just sending off (in two packages) the drives and the Master. The drives arrived ten days later and I am still (two months later) waiting for the Master. It has either gone completely missing in the Post Office somewhere or has been sent surface mail. After several phone calls and letters P&H have now promised to send me another one. By the time you read this I hope that I will, at last, be the proud owner of my own Master.

The moral of the story is that you do take risks when ordering stuff from overseas. Given the extremely good price ($1545) that Barsons is offering to User Group members for a Master that has already been converted to VHF and that has a three-month warranty, there may be little advantage in ordering from Britain.

FIRST IMPRESSIONS OF THE MASTER

Although my Master has not yet arrived, a Wellington member has kindly loaned me his (obtained locally) for a few days to have a play. I am favourably impressed. Basically, the Master has solved many of the things about the BBC that niggled. Most have been documented in the February and March Beeblet.

If those features are important to you, then it may be worthwhile upgrading to a Master. However, I must mention some disappointments. The first is that the graphics resolution was not enhanced. It should have ben possible to provide additional modes while retaining compatability with the old BBC modes. The second is that the existence of the real-time clock was not used to date stamp files under the ADFS. And the third is the lack of compatibility with software that used to run on the BBC.

In general, software will be incompatible if it uses fancy protection schemes assuming an 8271 disk controller, and if it assumes that the character definitions will be stored where they used to be. There will be other reasons. In general, Acornsoft has followed their own rules and most of their programs seem to run. A well-known notable exception is Elite, which produces the message "This program is only suitable for a BBC Microcomputer". Other funnies are Rocket Raid, which runs but is speeded up so that it is much more difficult to play, and Starship Command which works perfectly well except that the random hash when the scanner is not working is replaced by more structured stuff. Several notable non-Acornsoft failures are

The Music System, FRAK and Knight Lore.

Surprisingly, many sideways ROM images seem to run. I have not been able to try them all, but suspect that all Acornsoft ROMS (e.g., Pascal, Logo, LISP, Viewstore) will work. However, WORDWISE Plus fails - the Master comes up with "This is Not a Language". Strangely, the old WORDWISE works fine, but there is no way that it can take advantage of the shadow memory to use 80 column mode when previewing large text files. Ultracalc, Inter-sheet and Inter-chart also work, but can't make use of shadow memory. The 80 column mode in Inter-Chart is weird because of the character definitions, but 105 column mode is fine. Disc Doctor, Enigma and Toolstar also work, though many of their facilities will not be relevant to the Master. Failures include Commstar, Spellcheck, Sleuth, Dumpout, Murom, Help, Scribe, Printmaster, Starbase, Beebmon, and Oxford Pascal.

The included TERMINAL software includes an ANSI mode where it comes close to emulating terminals like a VT100. It provides hooks for, but does not emulate, the reverse video display of characters as on a VT100. Nor does it make proper use of the numeric keypad to produce escape sequences useful for DEC's editor EDT. These things are probably fixable and if no-one else does them, I will.

Documentation supplied with the computer is superficial. The WELCOME guide is fine for an overall description of the software, but lacks many of the details that are needed for more advanced features. For those you must pay extra for the Reference Manuals Parts I and II (15 pounds each in Britain; Part II is out and I have it), plus perhaps the Advanced Reference Guide, the View Guide and the Viewstore Guide. You do get enough information on View and Viewsheet from the Welcome Guide plus reference cards plus keystrip to do most things, though. Overall, it looks as if they saved money by bundling in the software but not the documentation for it.

The bottom line is, if you like BBCs you'll love the Master and want one. On the other hand, if you can already do what you want (most of the time) and don't need the extra features then it may not be worth it. If you only need a bit of extra memory and want to keep running the software you have now, then you should probably check out the sideways Ram and shadow boards now available (e.g., the very interesting Solidisk 256K board).

(AF: Acorn's Electron Pascal has been tried in another Master and found to work, as has Acorn's LOGO for the BBC)

EASTER

Anton Erasmuson

You've had the super version that does every year (AD that is); here is what Anton thinks is the simplest algorithm for Easter available, although apparently it isn't accurate for 1900 nor 2100 - they are NOT leap years.

Dates from this routine for Easter, Dominical date, and Leap year were checked for the period 1901 to 2025, and were found to be correct.

```
 10 REM ***Easter Date ***
 20 REM Anton Erasmuson, Jan 86
 30 REM Lilius/Clavius Algorithm
 40 REM Erasmuson formulation
 50 REM *** Gregorian Calendar
 60
 70 MODE3
 80 @% = &405
 90 FOR year% = 1970 TO 1990
100   PROCcalens
110   PRINT year%, er%,d1%, lp%
120   NEXT
130 END
140
150 DEF PROCcalens:REM   VALID 1901-2099
160 er% = 58 - (6+11*(year%MOD19))MOD30
170 IF er%>55 THEN er% = er%-1
180 d1% = 7-((5*year%-9)DIV4)MOD7
190 IF year%MOD4 = 0 THEN lp%=1 ELSE lp% = 0
200 er% = er%-(er%-4-d1%+lp%)MOD7
210 REM  er% IS MARCH date of Easter if er%<32
220 REM  else (er%-31) is the April date
230 REM  d1% is the Jan date of the first Sunday
240 REM  lp% is 1 if a leap year, else it is 0
250 ENDPROC
```

```
1970    29    4    0              1980    37    6    1
1971    42    3    0              1981    50    4    0
1972    33    2    1              1982    42    3    0
1973    53    7    0              1983    34    2    0
1974    45    6    0              1984    53    1    1
1975    30    5    0              1985    38    6    0
1976    49    4    1              1986    30    5    0
1977    41    2    0              1987    50    4    0
1978    26    1    0              1988    34    3    1
1979    46    7    0              1989    26    1    0
```

ACORN ADVANCED DISC FILING SYSTEM

Part III
Getting There

Tony Krzyzewski

Last month we looked at the initial concepts of sub-directories, this month we will look at how we reference those sub directories and the files they contain.

When we refer to a file it may no longer be sufficient to type just the file name, for example to load the SNAPPER program from last month's article by typing

```
LOAD "SNAPPER"
```

will not work unless the GAMES directory is the CSD. If we assume that we are currently in the ROOT directory we would in fact have to type the complete PATHNAME ie

```
LOAD "GAMES.SNAPPER"
```

A PATHNAME is in fact the route that the filing system has to search in order to find our program.

As another example, assuming once again that we are in the ROOT directory: if we wish to load the JAN program in the RECEIVABLE directory we would type:

```
LOAD "ACCOUNTS.RECEIVABLE.JAN"
```

From these examples we see three things:

1/ Directory and pathnames are separated by a . character. (full stop)

2/ The file name is always the last object in the pathname.

3/ All objects other than the last are assumed to be directories.

Let us now move out of the ROOT directory in to the ACCOUNTS directory. To do this you enter the command

```
*DIR ACCOUNTS
```

The ACCOUNTS directory is now our CSD

To Load the JAN program in the RECEIVABLE directory again, we would only have to type

```
LOAD "RECEIVABLE.JAN"
```

As you can see, it is only necessary to type in the pathname from where we are currently located. If we move in to the RECEIVABLE directory by typing

```
*DIR RECEIVABLE
```

it is then possible to load JAN by simply typing

```
LOAD "JAN"
```

If, while we are still in the RECEIVABLE directory, we wished to get a directory listing of the ACCOUNTS directory, we could do this by simply typing *CAT followed by the path we wish to follow. In this case the path is to our parent directory, so the command would be

```
*CAT
```

This type of referencing is known as RELATIVE as we define a relation to our current position.
It is also posssible to reference objects by ROOT RELATIVE REFERENCING, or in other words, going right back to the beginning and starting again!

For example:
If we are in the ACCOUNTS.PAYABLE directory and for a bit of excitement we want to play FRAK!, we can do it two different ways:

1/ Change directory to ROOT and load "GAMES.FRAK":

```
*DIR $
LOAD "GAMES.FRAK"
```

2/ Use the ROOT as reference:

```
LOAD "$.GAMES.FRAK"
```

Simply typing LOAD "GAMES.FRAK" from within the ACCOUNTS.PAYABLE directory will result in the message FILE NOT FOUND, mainly because it just isn't there.

As another example, if we want to find some information about the ACCOUNTS directory while we are in the PAYABLE directory we can do it in two ways:

1/ RELATIVE

```
*INFO
```

(Remember ACCOUNTS is our PARENT)

2/ ROOT RELATIVE

```
*INFO $.ACCOUNTS
```

It is possible to copy files between directories simply by specifying the pathname for the source and destination.

To copy FRAK into the ACCOUNTS directory while in ROOT:

```
*COPY GAMES.FRAK ACCOUNTS.FRAK
```

To copy FRAK into ACCOUNTS while in ACCOUNTS:

```
*COPY $.GAMES.FRAK @
```

Note that we had to use ROOT relative here as GAMES is not on the current branch of the directory tree. @ is an abbreviation for the CSD

ANSWERS to last month's questions:

1/ There are 2 sides x 80 tracks x 16 sectors on the disc;
Answer = 2560

2/ The parent directory of GAMES is the ROOT ($)

3/ The parent directory of PAYABLE is ACCOUNTS

4/ SNAPPER is in the GAMES sub directory

This month's questions:

1/ I am in the ROOT directory. How do I change to:
  a) the GAMES directory
  b) the ACCOUNTS directory
  c) the PAYABLE directory (in one step)

2/ I am in the GAMES directory, how do I load MYFILE?

3/ I am in the ROOT directory. How do I load
  a) JAN in the PAYABLE directory
  b) JAN in the RECEIVABLE directory
  c) FRAK
  d) MYFILE

4/ I am in the ACCOUNTS directory. How do I load

- a) HISFILE relatively
- b) HISFILE ROOT referenced
- c) HOPPER

5/ I am in the PAYABLE directory. How do I copy
- a) SNAPPER in to this directory
- b) JAN into RECEIVABLE

Next Month: The Library, and Wildcards.

MACHINE CODE PRINT

The Centaur

Unless you are a competent (not mostly harmless) programmer, putting a PRINT command in machine code is not easy. Whilst strolling through the ROM the other day, I found an easy way:

```
10 P% = &2000
20 [
30 LDA #1
40 JSR &FA49
50 EQUS "This will be printed"
60 EQUB 0
70 LDA #65 \ or whatever the rest of the program is
80 RTS
90 ]
```

The routine at &FA49 takes all text after the JSR &FA49 and prints it until it meets a Zero byte. Then it resumes running the program in the normal way. It is quite legal to place carriage returns in the string.
You could, for instance, insert:

```
  45 EQUD &0D0D
```

In fact any VDU codes can be used, on condition that they don't use a zero.
VDU28,5,20,30,12 is legal:

```
  45 EQUD &1E14051C
  46 EQUB &0D
```

but VDU19,4,7,0,0,0 is not legal:

```
  45 EQUD &00070413
  46 EQUW &0000
```

SOMETHING BIGGER II

Anne Flower

Last month I left you staring at a screen with Pacbeeb threatening to chomp through your memory. If you haven't died of worry, here is your chance to show him the way out.

By the way, this program was tested on BBC model B with and without co-6502, an Electron and a Master. Where there was conflict between Basic I and II facilities, Basic I was assumed. So: if you don't know what you've got, it doesn't matter!!

PROCget_move is the procedure that gets the player's move from the keyboard and processes it.
R% is the variable which will show what key was pressed. The program keeps GETting a keypress from the keyboard until the key pressed has an ASCII value greater than 135 and less than 140. The *FX4,1 in PROCsetup allowed us to get these values out of the cursor keys. Any key that is pressed will be ignored unless it is one of these 4 keys. What action is taken now depends on which key (up, down, left, right) is pressed. Hence the four IF statements. The program tests the first IF statement. If it isn't what is wanted, it automatically moves on to the next program line.

```
970  VDU31,OX%,OY%,17,2,237
```

At the position Pacbeeb was at before you pressed a key (OX%,OY%), change to colour 2 and print CHR$237 to overwrite Pacbeeb with something to show an eaten-out chip.

```
980  VDU31,X%,Y%
```

Move the screen cursor to the position that the player has indicated with his keypress. (This cursor is invisible thanks to line 370 - see last month). This doesn't actually print Pacbeeb at the square the player has moved him to (yet) - there is some other work to be done first, and the delay will be invisible to the player.

```
990  OX%=X% : OY% = Y%
```

This defines the "old" cursor position as being the same as the current one. The next move the player makes will use these new values at line 970.

```
1000  test% = FNfx(&87,0,0) AND &FF00) DIV &100
```

Gulp! In fact this is a "legal" way of finding out what the cursor is sitting on - which character on the screen. Although you can 'peek' memory addresses directly, this method will work irrespective of the Mode you are using, whereas the place in

memory to peek will vary from mode to mode.
Remember we haven't actually got round to printing Pacbeeb on the screen at the square he's just been moved to by the player. So the computer here looks at its map of the screen memory, and finds out what character (representing 1,2,3 or 4-byte chips) is in the same place as the cursor.
test% is the ASCII value of the character at the cursor position.

```
  1010  IF test% >224 AND test% < 229 THEN memory% = memo
ry% + (test%-224)
```

memory% is the variable keeping track of how many 'bytes' have been eaten. Pacbeeb can land on 6 different squares. 4 are those representing chips, one is an eaten-out chip (can't eat what isn't there), and the sixth is the exit door which is also inedible.

SO if Pacbeeb has landed on a square displaying a character in the range CHR$225-228 (ie greater than 224 and less than 229), then the amount of memory eaten has to be increased by the chip value of that square.

```
  1020  VDU17,2,shape%
```

At last we can print Pacbeeb at the square the player moved him to.
Compare:

```
   970  VDU31,OX%,OY%,17,2,237
            with combining lines 980 and 1020:
        VDU31,X%,Y%,17,2,shape%
```

line 980 moved the cursor to the desired position, but printing Pacbeeb at this position was delayed until line 1020.

Where did shape% come from? What is it? It's Pacbeeb, facing the direction he was moved - if the left arrow was pressed, he will be facing left, facing downwards if the down key had been pressed etc.... This was decided within lines 930-960. More on these later.

```
  1030  VDU5:MOVEX%*32,1023-(Y%*32):VDU18,0,1,shape%+1,4
```

So far we have been using text characters at text positions for screen display, and using the text cursor for positioning. However when you print a character on top of another character, it completely obliterates the previous character. Each character is made up of an 8x8 matrix, and printing a character plonks a whole new 8x8 on top. To add Pacbeeb's mouth to Pacbeeb however, we still want to be able to see Pacbeeb himself which this method won't allow.
By using the graphics cursor instead of the text cursor however, we can put a character on top of another without loss of detail,

because only the part of the 8x8 character that is foreground colour is put on the screen. As an analogy, if you are typing and make a mistake, you can go back and type a new letter over the previous wrong one. Both letters will still be visible. This is like the graphics cursor; the text cursor is more like a dab of twink over the wrong letter before putting the right one down.

VDU 5 is the command to print text using the graphics cursor
MOVE X%*32, 1023-(Y%*32) is how to calculate the graphics co-ordinate that matches the text co-ordinate (Print Tab), and you are MOVEing the graphics cursor to that same position.
VDU18,0,1,shape%+1 is the graphics equivalent of the VDU17s we have been doing. I defined mouths for each direction using the CHR$ value immediately after the one for Pacbeeb's body, so to add the mouth is simply a matter of adding 1 to Pacbeeb.
VDU4 is the command to revert to the text cursor (cancel the VDU5).

<u>PROCleft, right, up, down</u>
all do the equivalent depending on which was selected for action back in PROCget_move:
First they each check to see that the player hasn't tried to move off the "board". If he is, it beeps at him (VDU7)
It then defines shape% so lines 1020-1030 can print Pacbeeb facing the correct direction.

```
  1260  DEF FNfx(A%,X%,Y%) = USR&FFF4
```

This is used by line 1000, and is a version of the OSCLI call that is only implemented on Basic II upwards (which not everyone has). X% and Y% here are automatically treated by the computer as being LOCALised to this function by virtue of their being specified as parameters in this line, and bear no resemblance to the X% and Y% we have been using so freely elsewhere.

Well, that's the game; the listing after this carries on from where it left off last month. It includes the final part of the program not yet covered in the text above - that will keep for next month. Meanwhile, go back to line 10 and change it to a more useful version:

```
   10 ON ERROR PROCfinal:REPORT:PRINT" @line ";ERL:STOP

  880 DEF PROCget_move
  890 LOCAL R%
  900 REPEAT
  910   R% = GET
  920   UNTIL R% >&87 AND R%<&8C
  930 IF R% = &88 THEN PROCleft
```

```
 940 IF R% = &89 THEN PROCright
 950 IF R% = &8A THEN PROCdown
 960 IF R% = &8B THEN PROCup
 970 VDU31,OX%,OY%,17,2,237
 980 VDU31,X%,Y%
 990 OX% = X%: OY% = Y%
1000 test% = (FNfx(&87,0,0)AND &FF00) DIV &100
1010 IF test% >224 AND test% < 229 THEN memory% = memory%
+ (test%-224)
1020 VDU17,2,shape%
1030 VDU5:MOVEX%*32,1023-(Y%*32):VDU18,0,1,shape%+1,4
1040 ENDPROC
1050
1060 DEF PROCleft
1070 IF X% > 0 THEN X% = X%-1 ELSE VDU7
1080 shape% = 229
1090 ENDPROC
1100
1110 DEF PROCright
1120 IF X% <39 THEN X% = X%+1 ELSE VDU7
1130 shape% = 231
1140 ENDPROC
1150
1160 DEF PROCdown
1170 IF Y% <30 THEN Y% = Y%+1 ELSE VDU7
1180 shape% = 233
1190 ENDPROC
1200
1210 DEF PROCup
1220 IF Y% >2 THEN Y% = Y%-1 ELSE VDU7
1230 shape% = 235
1240 ENDPROC
1250
1260 DEF FNfx(A%,X%,Y%) = USR&FFF4
1270
1280 DEF PROCfinal
1290 VDU4
1300 *FX4
1310 CLS
1320 VDU23;11,255;0;0;0;
1330 COLOUR3
1340 kilo%=31-(memory% DIV 1024)
1350 byte% = 1024 - (memory% MOD 1024)
1360 PRINT'"BBC Computer ";kilo%;"K ";
1370 IF byte% < 1024 THEN PRINT;byte%;" bytes" ELSE PRINT
1380 PRINT'"BASIC"
1390 PRINT
1400 ENDPROC
```

ELECTRON COLUMN

Steve Cosgrove

BACK to the basics in life. Changing nappies, and obtaining extensions for projects. One tends to lead from the other - I mean the longer an extension you get before handing in a project, the more other things you can fit into 24 hours. If not careful, nine months later, you can be changing nappies.

Our situation was not quite as haphazard as that, and obtaining extensions was made necessary by spending too much time changing nappies. (Particularly at 3.00 am.) Life is settling down though. We are now in the University break, so I can catch up on overdue projects, and then (in some cases very) overdue correspondence. I am not too far behind though, and any more letters will be answered too.

GAMES:

Relaxing can be a very rewarding past-time. Some people find sitting down behind a wizz-bang pop game on a computer to be relaxing. Personally I find something like Othello (Acornsoft "Reversi") more appropriate. There are many games of both catagories available for an Electron. Many games ("Guardian", "Meteors", and some others) operate at the same speed as the BBC versions. Some, such as Monsters, run much slower, and often suffer an even greater loss in sound quality. There are also those ("Snapper", "Hopper") which were slowed down when implemented on the BBC. Their Electron versions appear to run at the same speed as the BBC versions, but only on the machine each was intended for. I have not seen anyone who can get past the first screen on "Electron Snapper" when it is run on a BBC!

Why is all this so? Games are often slowed down to what the Author considers a manageable speed. There are a wide variety of different ways in which the timing can be controlled in a game. One is a simple "wait loop". In this case, the computer sits around doing nothing for a few hundreths of a second. Most general operations take longer on an Electon than on a BBC. The Electron even takes longer to do nothing! (Doing nothing has a whole new meaning to a computer.) So games slowed in this way will run more slowly on the Electron. The slowdown applies to both movements of hairy monsters or whatever, and the "waiting" between movements. Games written in BASIC suffer markedly from this problem. It also accounts for the latter two categories I mentioned. Sound on the Electron is limited to one note at a time. The BBC can play three notes in harmony. This accounts for poorer sound on Electron games. Games which run at the same speed on either computer tend to do very funny things to the screen. The details are very complicated, but suffice to say that each computer updates - "refreshes" - the screen at the same speed (50 times per second). Some games use timing based on this "refresh rate", so appear to run at the same speed. In practice the BBC will be "waiting" for the next refresh (one fiftieth of a second is a long time) while the Electron will only just get everything ready in time.

What to buy? Anyone who has compiled a list of games that do or do not run on the Electron would be welcome to send it in to be published in our magazine. Otherwise the idea is that if a game does not specifically say it will run on the Electron, try it out first. There are very good "Electron Versions" of some games (eg "Elite", "Citadel"). Absolute speed freaks with a BBC might even like to try an Electron version of a game - to add a new challenge.

For now I will keep on changing nappies, completing overdue assignments, replying to letters, and generally keeping up with other necessities of life. Including playing Othello - of course.

FAULT-FINDING

ANALOGUE PORT

The socket that takes joysticks is, on some BBCs, screwed into the printed circuit board. However it hasn't been done to all, although provision for firm fixing is made. Instead the socket and wires to the pcb are just there. What happens then when kids haul on the joysticks, grabbing them off each other, pulling on the cable, stressing the socket?? Eventually the wires from socket to pcb break under the strain. Joysticks become unreliable and the kids take to pounding the keyboard instead. John Andersen discovered this, the wires were re-soldered and the socket attached firmly to the pcb, and the kids are off the keyboard and using joysticks again.

SOLIDISC:

Another of John's problems - fixed by Mike Hoult. His 128K sideways RAM wasn't quite behaving itself. Besides soldering in these cards, he found it necessary to shave a bit off the edge of the perspex function-key strip holder where it fits inside the computer to stop it and part of the card meeting.

SIDEWAYS DISCS

If you're worried about the potential for losing discs of software, there is one thing that will put your heart in your mouth - the child who forgets and puts a disc in sideways: not just upside down or back-to-front, but actually sideways!! Don't panic - two households where this has happened agree that they have been able to extract the disc (with some judicious wriggling), and the disc and drive have neither been worse for wear. Not that it is recommended practice mind you, and in one instance the disc wouldn't work immediately but was fine by the next day.

LOGO BOOK and the RUSSIAN DATABASE

Readers of the October 'Beeblet' article on Logo may recall a plea for the return of Papert's "Mindstorms" which had been lent to someone but the author couldn't remember who! It was returned through the mail with this apologetic note:

The book would have been returned earlier, as I am most meticulous in these matters, and have all the books I have borrowed on a large computer data-base. Lucky me.

Unfortunately, the data is entirely encrypted in Russian, as the system is second-hand Government surplus, and a sad tribute to the 'lowest-tender-wins' principle. Luckily, the local agents had a disaffected Croatian Nationalist who had learnt Russian at some place called the Gulag Archipelago, and he has written an interpreter. Unfortunately, he didn't speak English well, except for a rather limited vocabulary, mostly adjectives, normally terminated by the phrase "sodden Russians". Luckily, he was very adept at using a rather old tourist phrase book, filled with endearing ejaculations such as 'Long live Emperor Franz-Joseph'. Unfortunately, he had to be rather artistic in his estimation of modern terminology, and 'computer bug' was rather freely translated as 'calculating bed-louse'.

The ultimate result of this confusion is that:
'MINDSTORMS' was recorded as "TAKE CARE OF THE WOLF WEATHER"
'PAPERT' became "FATHERSAUCY"
'FLOWER' became "DAFFODIL or RIVER?".
The borrowing date, however, was exactly right, after making allowances for the Russian Orthodox Calendar. Please accept my profuse apologies for the delay. I also apologise for writing this by hand, but I have had some disturbing problems with my word processor. (ex Govnt stock) Regards, VR0104

Well, after that, what can any editor do, but demand an article on data bases......

After commissioning an introductory article, I've run out of space this month to print it!

A PRESSING PROBLEM

YET ANOTHER TRAP FOR BEGINNERS - Disc Fault

Peter Osborn

Excitement was mixed with not a little apprehension when I received my B+ direct from England. Would it still work despite the attentions of the postal services of two countries? Things looked good, the parcel didn't have a "FRAGILE" label on it - they're no fools at Watford, why look for trouble; and clearly Port Agriculture in NZ had not sterilized it as the plastic hadn't melted - probably got just a quick dunk in formalin.

I'd picked up a Shugart disc drive locally secondhand - it came in a natty little custom-built alloy box under a plywood shelf - and so with a monitor from the User Group I thought I was all set to go. I was wrong. When asked to load anything, almost invariably the message "CRC error 00/00" appeared and everything stopped. A novice, this meant little to me, but when an expert friend also confessed puzzlement, anxiety grew. What had I thrown my money away on, the drive or the B+?
Systematically swapping bits over with a known working setup showed all the components worked perfectly - individually that is - but not as a whole, and a borrowed Mitsubishi drive gave no problems for a week. We were no closer to an explanation when a chance to swap the Shugart for a Mitsubishi came up. Since this would avoid (rather than solve) the problem, this opportunity was seized with somewhat impolite haste. The Mitsubishi came out of a double unit and therefore was without a case. It worked beautifully like that with my B+ - then I put it in the custom-built alloy case under the shelf and got "CRC error 00/00" again. I started to take the setup apart, took the monitor off the shelf but then got called away for an hour or so. When I got back I thought I would try it once more - and it worked - drive and computer perfectly attuned. So I put the monitor back on the shelf and tried again. "CRC error 00/00" again. Monitor off, it worked fine; monitor on shelf - error message.

Novice I might be, but even I recognise conclusive results when I see them. A very close inspection of the alloy box/shelf unit revealed very little clearance between the moving bits of the drive and the underside of the shelf, and the weight of the monitor on the plywood shelf was clearly enough to cause some sort of misalignment. I bought a proper case for my Mitsubishi, used the plywood shelf to start the first fire of winter, and have been computing happily ever since (as has the guy I swapped the Shugart with).

ELECTRON VIEWSHEET INDEX

Compiler J Norman

A
Absolutely 20
Auto entry mode 16, 10
B
Bar chart display 62
C
Copy 11, 106
COL 24
COLUMN HEADING 38, 99
Conditional
  operators 69
COMMAND LIST 91-95
CREATE 59
D
DELETE 7
Deleting 50
Deleting blocks 122
DISCs 134
E
Edit window 29
Edit slot format 48,123
ERROR list 140-1
ERRORS 107
ESCAPE 12
F
Formulae 109
Functions 111-116
FUNC A 98
FUNC C 98
FUNC D 5
FUNC E 5
FUNC F 98
FUNC H 49,101
FUNC I 50
FUNC J 49,101
FUNC L 13
FUNC M 50
FUNC N 50
FUNC O 50
FUNC P 12,105
FUNC Q 10,117
FUNC R 98
FUNC S 5
FUNC U 38,99
FUNC V 16,107
FUNC W 13,105
FUNC X 5
FUNC Y 38,99
FUNC Z 117
FUNC 0 11,106
FUNC 1 17
FUNC 2 28,131
FUNC 3 31
FUNC 4 12,106
FUNC 5 106
FUNC 6 106
FUNC 7 48,123
FUNC 8 26
FUNC 9 106,11
G
Go to slot 26
H
Headings 38,99
  Turn off 31,39,99
  Turn on 39,99
  User defined 99
I
Inserting
  Rows 50,102
  Columns 50,102
  Limits 103
L
Labels 104
  justification 104
  entering 10
LOADing 52
Linking file 59,134-7
  READING 60,135
  CREATING 59,134
  WRITING 60,135
LOOKUP 70
M
Mode 4
N
Next window 31,128
NEW

- Number range 124

O
- Operators 110
  - precedence 111

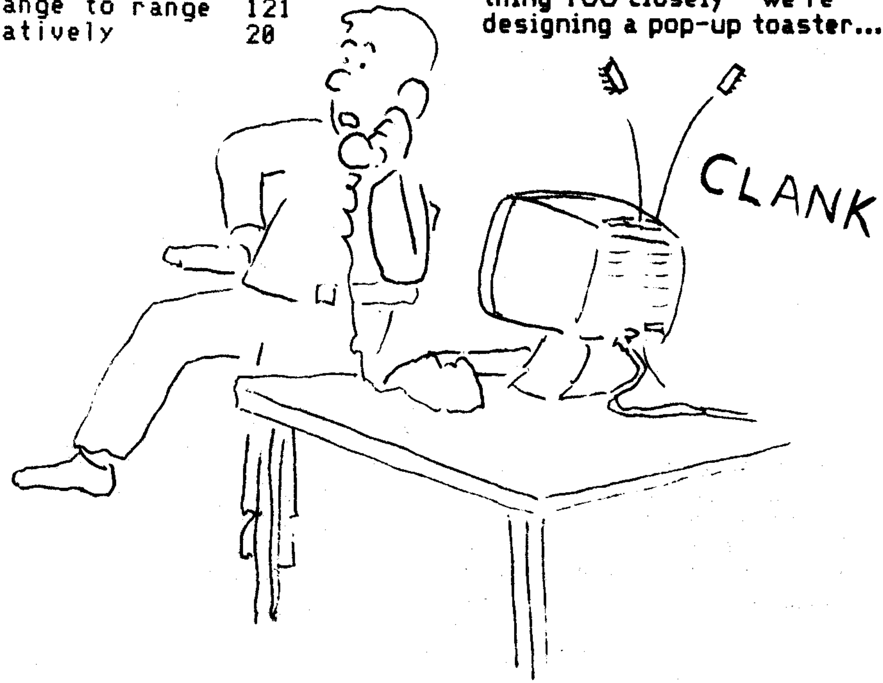
P
- PRINT 40,130
  - Printer driver 129,132
  - *FX8 40,130
  - Windows 41,131
- Printing out 40,129,132
  - *FX5,0 40,130
  - *FX5,1 40,130
  - *FX5,2 130
- PROTECT ON,OFF 49
  - ROW 49
  - COLUMN 49
- Protection 49,100

R
- RECALCULATE 9,117
- Ranges 16,110
- Replication 17,118-122
  - two dimensions 22
  - slot to slot 119
  - range to slot 120
  - range to range 121
- Relatively 20
- ROW 24
- ROW HEADING 38,99
- Recalculation 117
- RETURN 139

S
- SAVING
  - disc/tape 37,51
  - window defns 53,132
- SCREEN 44,133
- Screen window 125
- Slots 10
  - editing 11,46,105

V
- Values 104
  - entering 7
- Value formats 123
- VIEW 133

W
- Windows 25,125
- WRITE 60



...no, no Jim - it's the CAD program emulating the real thing TOO closely - we're designing a pop-up toaster...

MODEM OWNERS LIST:

AUCKLAND	(std 09)
John	535-4780
Gary	444-7907
John	266-8275
Graeme	585-796
Jo	814-9623
Mike	266-5640
John	865-233
Malcolm	bus 416-3000 x984

HAMILTON	(std 071)
Ken	54-779
Neil	393-876

TOKOROA	(std 080)
Don	67-690

NELSON	(std 054)
William	76-753

CHRISTCHURCH	(std 03)
Carl	228-370

SHEFFIELD	(std 051)
Helen	638-860

TIMARU	(std 056)
Lloyd	61-412
Stephen	81-761

WELLINGTON	(std 04)
Ross	780-157
Mike	785-437
John & Esther	793-292
Ceri	793-731
Simon	753-349
Colin	366-170
Alan	838-899
Gerald	757-909
Tim	278-502
Anton	286-289
John	838-715
Steve	687-558
Bruce	861-662
Don	848-235
Neil	338-450
Jean	338-416
Keith	792-990
John	289-312

INVERCARGILL	(std 021)
Trevor	57-644

DUNEDIN	(std 024)
Llew	741-284
Alan	730-257
Martin	42-031
David	770-051

(AF: Hope these all got typed in correctly; please check if your number is wrong, and let us know)

POST QUESTIONNAIRE

Several members offered programs that might be too long for a magazine; if they send them in, we can build up a library for members.

The User Group sells discs to members. The organiser is prepared, when selling them, to supply items on them from the library at an extra charge, but needs your programs to be sent in before he can pass them on...

ACORN SPEECH

Tim White

If you have had the speech generator chip fitted, and have small children, the following program may be of use to you. The operation of the program is quite simple, and my 2 yearold daughter learned the alphabet in a week from it. It is still her favourite.

The program waits for a key to be pressed, prints out the corresponding letter on the screen, and causes the synthesiser to say the letter or number, using the SOUND-1,G,0,0 command. If Return is pressed or more than 37 characters have been printed, the cursor is moved down two lines and a colour code is printed, so each new line is a different colour.

In the original program, a problem arose because many of the sounds are of longer duration than the auto repeat period, so that it was possible to fill up the speech chip's data buffer. If you kept your finger on the Return key too long, you had to endure a couple of minutes of a sound that I can only describe as a magpie with a sore throat flapping round in a dust bin!
The solution is to reset the speech chip after each key press. Why Acorn don't give this command in the manual I can't imagine. It is:

```
*FX159,0,240
```

By holding a key down you can now make the computer stammer, and by altering the period of the auto-repeat, you can get some interesting effects.

```
   10 REM speak the alphabet
   20 REM with Acorn speech chip
   30 REM by Tim White
   40
   50 MODE7
   60 *FX200,1
   70 I% = 1 : J% = 1
   80 REPEAT
   90   G = GET
  100   I% = I%+1
  110   *FX159,0,240
  120   IF G = 13 OR I% = 38 THEN I%=1:J%=J%+1:VDU10,10
,13,128+J%MOD6 ELSE VDUG
  130   SOUND-1,G,0,0
  140   UNTIL FALSE
  150 END
```

ELSEWHERE IN THE HOUSE...

Sadie

In case I left you wondering about the tangle of cable round the house for the beastie to communicate with the computer, it actually uses an infra ray sensor. There are several ray senders through the house now. About the only place it can't cope is under the double bed! The beastie can also detect which room it is in (handy when it has been deactivated for a while), and is capable of changing over some of its own battery packs.

Most of the kids have developed affinities with particular computer languages and use them exclusively. For instance, hubby wrote the Ferris suite in machine code mostly, but:

Video is in LISP - it gets the beasty to go over to our old and battered video recorder and get the appropriate programme recorded. Practically every tape in the house now shows at least one episode of Fred Flinstone!

Pickup is in LOGO - quite a few kids enjoy using beasty plus a pair of Voltmace joysticks with keypads to instruct it to pick up some item from the floor in one room and take it to the toy box / rubbish bin / wardrobe to tidy up. It gets competitive, with children vying with each other to see who can pick up the most the quickest, so the program has a stopwatch built in too, and prints out the timings! I can now laugh at the memory of the time one tried so helpfully to get beasty to take the ashes from the grate out to a bucket for future disposal!!

Knit is in BASIC, and designs knitting patterns for the electronic knitting machine. It's rather more than a gigantic user defined character generator, as it also recalculates patterns so I can use different thicknesses of yarn than the original pattern might call for.

Hitide is a FORTH word much used by one who hates cleaning the bath after using it. She's trying to get me to replace the taps with easier-operated arm ones, so beasty can fill the bath as well to a predefined depth. I've fobbed her off - until beasty can get the plug in properly with one command. Then I have to worry about the effects of mixing water with electricity.

House contains the floor plans, and is also in FORTH. It's the beasty's road map, not an architect's.

Hubby says the Music 500 uses a language called AMPLE. I just looked at his waistline and suggested he was already, and wasn't there some language called WHITTLER or EXERCISER he would be better off using. HE took that as licence to buy some body building packs from Micro User!!

We have had to ration computer use - pre-beastie it was hubby's exclusive domain. Now he is relegated to 2am-5am. The kids are allowed only 1 hour per day each (non-cumulative), so why should he complain, when he has thrice as much time as anyone else? That teething and tummy bug troubles usually erupt in this period is only coincidence, but since he's up and awake anyway...

When do I use it? Hardly ever. I just tell the nearest person what I want done next (within the hours 6.00-9.30am and 4.00-8pm), and get on with the housework. I do have plans for further use though... all that repetitive scrubbing action on walls and ceilings would suit a heavier-duty robot arm, and might even be able to cope with exterior weatherboarding. I don't care what language it is written in so long as it works. I'm a little suspicious of the child who insists it's a natural application for a Z-80 co-processor - sounds a little expensive to me, seeing what's being done with what we've already got.

I've just about thought through a 'Laundry' suite, and it's going to be a beaut:
First of all the computer will dial up the Met Office for the weather report, and use its speech decoder to decide whether it will be fine enough to use the clothesline. It could access a computerised database, but since that costs money, and there's already a speech decoding unit ("Big Ears") around, it can jolly well learn to cope with the peculiar accent the Weather Office uses. It would then decide if it was fine enough, and if there was enough daylight left to warrant pegging the clothes on the line. (I thought Beeblet could publish a program to produce sunrise & sunset times that we could use) Otherwise it will then get the robot to transfer the clothes from washing machine to dryer and dial the correct time and temperature to get them dry. One of the children is still somewhat uncoordinated, so I'm working on that child to program the movements required to select an item of washing, shake it out, then peg it up properly...

The only thing is, how does a "meccano arm" know if it's got a shirt or a sock to deal with - or will it lift up an item for a child with keyboard to identify the item for it? Perhaps the whole Play Centre could meet at our place each wash day, and have fun identifying garments for the beastie.

ADVENTURES:

SPELLING:
Sometimes there are spelling errors in adventure games, which can irritate those who were born with the dictionary in their brains, but there is one apparent one that isn't:

Level 9's SNOWBALL has a "machanical Nightingale" who is used to dealing with "psychotic" humans. That's a very clever way of combining "maniacal" with "mechanical" to get the two meanings across at once.

MY SNOWBALL ADVENTURE

Liz Metcalfe

As I jumped out of my coffin reinforcing my suspicion that I was in fact alive, I found my first game of Snowball to be a short one. I looked around my general vicinity to accustom myself to the surroundings. It was then the hideous nightingale appeared and murdered me in cold blood - well cold oil maybe since robots don't have blood.

Once I got good at the game I found there was a multitude of ways I could die. I had the choice of being knocked off by the nightingale, of running out of air and choking to death, of being lost in the darkness of space never to be seen again, of being cooked to death by a laser beam and even of falling to my doom from a great height. Just a few of the exciting moments I have died through.

As some of you may well know, some games become addictive and this one certainly did. I would wake in the middle of the night with wild dreams of where to stick the diagnostic probe. There were moments at work when through the corner of my eye I could have sworn I saw a butlerdroid appear playing a flute while chasing a cute little pussy cat. Some cute pussy cat - you wouldn't believe what it did to my credit card. No I'm not going to tell you and take away all the fun. You are going to have to get friendly with the purrer from the Black Lagoon.

After weeks of hard work and torture I finally won the game and to my amazement it had the cheek to only award me 950 out of 1000 possible points. So here I am back at the beginning retracing my steps and wondering what is the meaning of life.... Life, don't talk to me about Life.

ADVENTURE GAMES - AGAIN

Bryan Halliday:

*** Introducing PAS the wonder games machine ***

GCF has been at it again. He has refined further his concept of an adventure games program. This time the menus have been abandoned in favour of command input. The player directs his hazardous trip through the adventure with English language commands. Although not yet available, this is a preliminary look as it develops.

NEW FEATURES:
Hidden from the players are a number of refinements. The inhabitants of the maze have taken on a new life and objects have gained new attributes. Any animate creature can be made to move around following a set of rules, eg head north, or make random moves or both. Movements take place in step with the player's moves - this is not a video game where hesitation allows the enemy to get ahead.

Objects can be invested by the game writer with several attributes. For instance the attribute of holding other objects or holding liquids. A bag can hold other objects but a sword can not, a bottle can hold liquids, and a string kit cannot. Once given, the program enforces these attributes.

As well, travel has become easier. Ask to go north and the machine finds the nearest place due north, checks for any magic spells or road blocks, and ZAP you are there.

Except for the ZAP part (adventurers with fallen arches are not encouraged), this is nearer to real life. Animate creatures can move about, scissors can cut paper but not rocks.

WORKINGS
The method of working is similar to SAS (September '85 Beeblet). The machine goes through a cycle of:

1. Describing the current state of the room and contents
2. Accepting and actioning the command input
3. Reporting any consequences of the command.
4. Before starting the cycle again any automatic movements are done

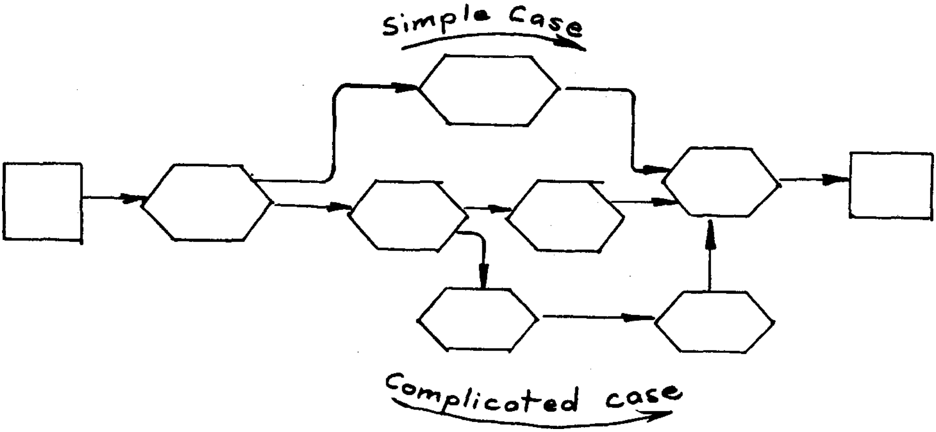
In the usual computer way it stonewalls if it can't understand the input, and you have to rephrase your command to get any further.

HOW DOES PAS WORK

The PAS game playing process uses a method that separates the presentation process of the program from the details of what is in the adventure. The writer of the adventure supplies data that describes the scene and inhabitants of the adventure world. When the game is played a program uses the data to create the adventure.

PAS has three different types of data statement so a game consists of four components:
1. The program (or Engine) that directs operations
2. Data for the objects in the game
3. Data for the commands recognised
4. Data for the scene of the adventure (rooms if underground).

The adventure writer can modify or replace any of the data statements without altering the program.



Because the player is in only one room at a time and all of the rooms are different, the data statements for each room are held in a small group called a frame. The frame holds the room description, rules for entrance and exit, warning messages if the rules are broken, etc.

The engine's job is then a simple repetitive process on the data statements. At the start of the game positions of objects are put into tables. Then for each cycle the engine:1. looks up the appropriate frames and prints out the descriptions
2. gets a command from the keyboard
3. actions the commands
4. prints any messages about the command
5. if the player was moved find the next frame.

Before the next cycle the engine checks all the mobile objects and moves each according to its rules.

LANGUAGE HANDLING

Also like SAS the deviser of the adventure (writer if you like) writes the adventure instructions in a language of GCF's devising. It is similar in having two-letter commands and has many common commands, but has been revised and enlarged.

The main change is the inclusion of a number of language handling commands. The good news is you have control over how the sentences are interpreted and acted on. The bad news is that the command interpretation is not automatic but has to be written by the game writer. Using frames reserved for language handling you can test your ingenuity in interpreting the input and initiating action in the adventure. This is definitely another dimension, you can join the thousands of researchers around the world who have thrown millions of dollars away in attempts to understand how a language like English works.

In this case the problem is not quite so great. The objects in the adventure are listed inside the program, actions are so restricted, that any practical input will be a terse command.

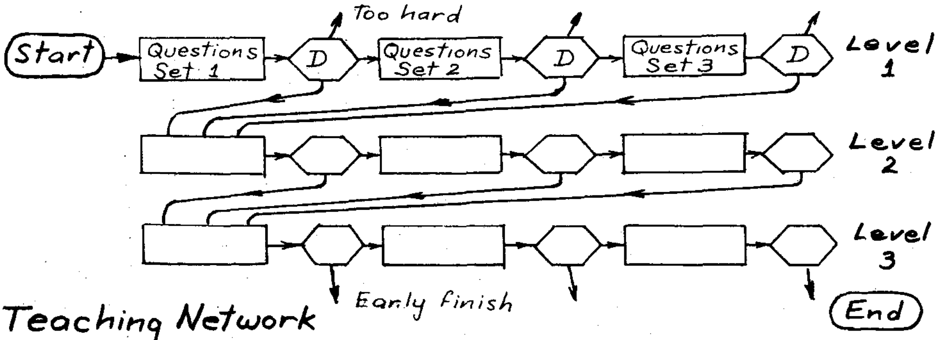
THE REALITY

OK! That's the concept, what is the reality. There is a program. Most of the time it works, and quite a lot of the time it understands the player's instructions. GCF is still polishing and testing it so it won't be available for a while. Also its expanded language is a bit of a hurdle for a new user, the quantity of commands available to an adventure writer and the 2-letter abbreviations used make starting up hard. A decent manual is going to take a while to prepare!

WHAT IS IT?

Curiously what has been produced is not necessarily an adventure game. The concept of moving from frame to frame, and the term "frame", were originally set up for what is now known as "Expert Systems", a branch of Artificial Intelligence research. I don't claim it can be used as an A.I. program (Challenge!) but there are a few simpler things it can be used for:

1. A teaching program
with multiple levels of program and several sets of questions at each level. Progression from level to level can depend on answers to questions:



Teaching Network

For practice "D" = Decision : more examples or next level
For teaching "D" = Decision + Furthur explanation text.

2. A simulation

Here it can be any process with discrete steps. As the movement between frames happens without a time scale, it is probably not useful for something like a train movement simulation. (Challenge again!) However for simulating shifting a flat using two cars and a trailer it is probably ok. Also you can simulate the sequence of the goose, fox and bag of corn puzzle. Maybe you can get the program to solve that puzzle! After all its ancestors were A.I. programs.

3. A Complicated series of questions and answers

where some questions are skipped depending on previous answers. (see next diagram)

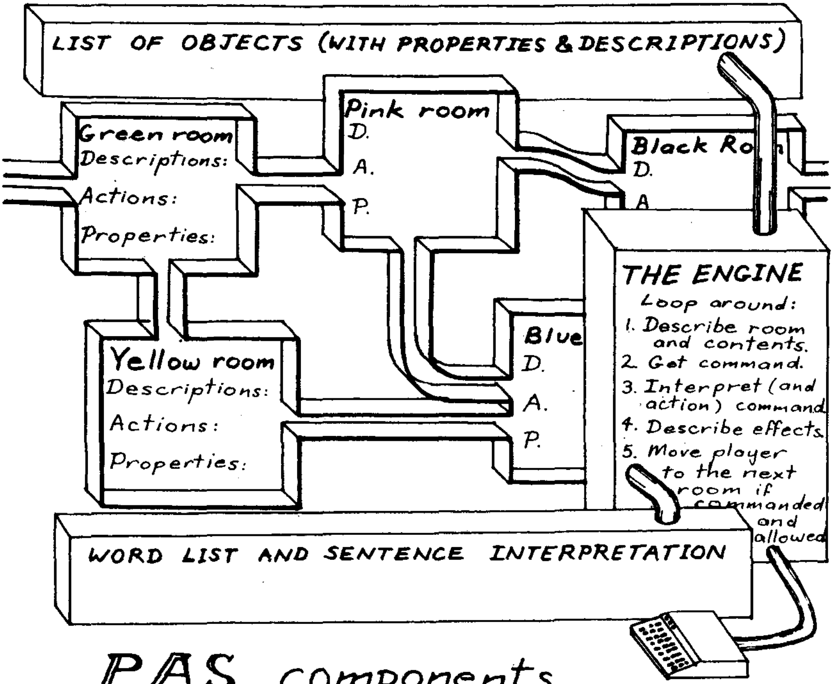
Well that's the short introduction to PAS. The subject will probably get more airing in future issues, looking at more of the details.

come from p.33:

systems operation is foolishly documented under CP/M Basic, with the consequence that I didn't find it, since I haven't any Basic programs that don't run on the 6502, and the need didn't arise.

The Z80 has served me well with quality hardware and software and still holds many surprises. In fact I haven't scratched the surface. It is quality, and if I have a regret, it is that MS-DOS (which is somewhat CP/Mish) is the new standard in business for both the world and Acorn since the Master.

Since it has served the purpose for which I bought it, the unit is for sale. I can be contacted through the User Group.


LIST OF OBJECTS (WITH PROPERTIES & DESCRIPTIONS)
Pink room
D.
A.
P.
Green room
Descriptions:
Actions:
Properties:
Black Ro
D.
A
THE ENGINE
Loop around:
1. Describe room and contents.
2. Get command.
3. Interpret (and action) command
4. Describe effects
5. Move player to the next room if commanded and allowed
Blue
D.
A.
P.
Yellow room
Descriptions:
Actions:
Properties:
WORD LIST AND SENTENCE INTERPRETATION

PAS components

NEWS FROM...

AUCKLAND:
At Adrian's work their BBC system includes an Adler typewriter and View. The secretary took 2-4 weeks to become really competent, with no prior computer experience, and some reservations when starting. Now she does very little without the wordprocessor. He asks:
What are Auckland doing for topics? and publication of this info could stimulate attendance.

WELLINGTON - p.39

ELSEWHERE:
??
Fill in this space for a start and post it in:

Z80 SECOND PROCESSOR

Jeff Hunt

Over a year ago I began a year's course in file handling using COBOL. Since it is not available on the standard BBC B a certain Mr. Flower suggested the purchase of a Z80 second processor which comes with a great deal of packaged software including the highly rated Compact Interactive Standard COBOL (CIS COBOL). The $1500 price tag was a hurdle but having already learned Pascal by correspondence and not enjoyed the experience I took the plunge. The advice I had that I could get by with one disc drive proved to be wrong and ultimately 'retooling' cost about $2000.

Events have overtaken the Z80 now but since very little has been written on the subject I offer this. My experience is limited almost entirely to the word processor and COBOL, and much of the packaged software I've never even got around to trying so my remarks will be limited.

The Z80 chip is important because it has been a trusty servant since the early days of personal computing and had written for it CP/M - a standard operating system which allowed compatibility of programs between the many Z80 machines. To enter the business market Acorn tapped into this store but since the 6502 is not CP/M compatible the Acorn Z80 was born.

The system comes with eleven books documenting seven applications plus CP/M itself and numerous support programs. The co-processor is a neat unit that sits beside the beeb and is activated after switch on by SHIFT-BREAK at which it asks for CP/M disc in drive A (that's drives 0 and 2 combined to a beeb B). CP/M is a disc based system and relies on discs for both the system itself and for numerous support facilities. Superb documentation makes the awesome task of familiarity more acceptable although with seven major business packages including database and accounting it's not a half-hour affair.

And so to specifics: I've not used any other word processing package and my remarks may be naive but it is a fairly powerful beast - beautifully documented both at keyboard and book level. Apart from the usual write, delete, move, justify type approach it has features such as automatic backup, a four document 'swap' file and double document handling. The swap file and backup facility make very difficult indeed to lose a document and also provide convenient access to the last four documents - after which they can be written to ordinary CP/M file and over written in the swap file. A feature which was very useful to me writing programs was that on loading, Memoplan (the w/p) accesses the last document and replaces the cursor where it last was. A great boon for debugging when your last clever change was to remove a bug but didn't and you're not in the mood to search through trying to remember what it was you last did anyway. A further perk was on exiting Memoplan the name of the file last written remains on the screen facilitating further file processing such as compiling.

The CIS COBOL compiler is good and well documented although it is slow both on compiling and running. In fact according to my brief check 5 minutes is normal for a fair size program compile, and execution is not only slower than a comparable BBC Basic program, but considerably slower than other Basics as well!! Perhaps this is not so serious for Cobol. Arcade games written in Cobol are fairly thin on the ground. (Z80/BBC Basic appears to be slightly faster than on a single 6502). Another quirk is that certain standard items such as VARYING AND OR and COMPUTE are missing and I missed them.

On the plus side the debugging aid ANIMATOR is supplied and this has to be seen to be believed. It is incorporated at compile, and at run time allows the program to be stepped through as slowly as required with the listing on the screen and the cursor marking the statement being executed. Furthermore any variables can be examined at any time. I couldn't quite shake the feeling that this was cheating and only used it when total confusion or depression took over.

As an aside on Cobol, I like it. Given a choice between the endless begins and ends and ('s and *'s and definitions of definitions, and ugly abbreviations of Pascal, or the complete opposite of the "We'll sort it out as we go" of Basic, I'll take the dignity of Cobol any time. The English is expressive so that a decent piece of Cobol is self-documenting and the infamous verbosity is all either to achieve this clarity or is used in the large data and environment divisions which lay the groundwork to ease the burden in the procedure division (where other languages can disappear into a morass of housekeeping detail). Any language that allows:

```
MULTIPLY QUANTITY BY UNIT-PRICE GIVING TOTAL-PRICE
```

with a full stop (not a b----y colon or a comma) to complete the sentence has to be looked up to. - No not because you've fallen over laughing!! I realise I'm not expressing a typical view, and look forward to being challenged by the professional programmers and alternate language buffs.

This article was written on Memoplan and the file will later be copied across - using the CP/M utility supplied - to Acorn DFS. Our illustrious editor will sort it out. The copying across

GOTO p.30

LETTERBOX

Associate Professor J.M.A. Brown

Dear Mr Wills,
Being myself Joint Chief Editor (also unpaid and overworked) of an international journal in a science subject I can appreciate many of your problems as the Editor of Beeblet and can sympathise about the many undoubted complications you face in your tasks.

However I must question what I consider to be a very dubious practice ethically - altering a submission and then publishing it without any further participation by the author. This happened to the article I submitted to Beeblet and which was published in the March issue. It is a recognised convention in technical journals that editors and reviewers have the right to review, reject/accept improvements to articles submitted for publication but NOT to interfere with them without the author's express approval. As an editor I have pencilled in 'improvements' and stated that the article will not be published unless they, or comparable alternatives supplied by the author, are effected. In every such case I have always been careful to instruct the publishers that the author must accept the alterations before publication. The need for this type of approach arises from the fact that authors are unpaid and own the copyright until the article is published (before publication authors must assign the copyright to the publisher). The formality of such procedures may be a little grandiose for a "club" magazine but I suggest that, if the practice of changing submissions without the author being further consulted is to continue, then this should be made clear by a statement to that effect published regularly in the magazine. The fact that Beeblet is on sale in some shops to non-members suggests that this caution could well be necessary.

Incidentally, there is no sense of "they've interfered with my brain-child" type of carping in all of this - I have completely lost interest in the sort ('improved' or the original)! Subsequent to its submission I modified the associated name entry program (enter, amend, add or delete) and patched in Ian Birnbaum's machine code alphabetical bubble-sort. It's so much quicker and efficient (and takes less memory if automatically prior-loaded!) for my work-related purposes that the original submitted to you is now relegated to what I term my "Disk for programs of historic-interest only"! (And there it will remain unless I run into problems with the location of the machine code on the Master - and that seems only too easy to do!

Whilst I regard the improvements made by Steve Cosgrove as valid in the purist sense, they have, in my opinion, obscured the value of the sort to those who use the computer as a tools rather than because of a prime interest in computer technology by way of programming skills. The sending of the article was triggered by a previously published piece in which it was clear that the author was even less expert than myself on alphabetical sorting and was indirectly making a plea for help (and please believe me when I say that I regard myself as very inexpert - but perhaps because of that more likely to understand the problems of others). I venture to suggest that most who are in the early stages of programming have much less understanding of TRUE and FALSE than they do of GOTO and might be unable to modify the program to suit their needs as it now stands whereas the 'mechanisms' more clearly revealed by GOTO could be comprehended relatively easily.

This tendency to go to fully structured programming and a greater insistence on purist criteria in computer magazines could be alienating many like myself who are primarily interested in what can be done rather than how to do it in the 'best' possible way. My only reason for venturing into the area of writing programs is that, having particular needs not adequately covered by published material, it is the only way to get what is required when funds are not available to pay someone else to do it. Certainly, in my own ignorance, I have had increasing difficulties in adapting techniques published more recently to my needs for the computer as a tool rather than as an instrument in its own right. The recent articles on "sorting" in Beeblet were infuriating in this respect by the way they presented Pascal structures and then left it to the reader to convert them to BBC Basic if the sort was required before the next article (more than one issue separated from the original). I do not have Pascal on my machine and have neither the time nor, I suspect, sufficient computer logic to do this sort of conversion easily. Thus, whilst there is definitely a place for this type of article for those who are interested in problems for their own sake, I would ask you to remember that many are not. Other magazines do the same type of thing - a recent rapid machine code sort for integer numbers in Micro User included the throw-off remark "that it could be easily modified for alphabetical sorting". Not by me for one, and there are probably many others with alphabetical sorting requirements who are in the same position.

Thus, I suggest that if articles like the original one I submitted are seen to be useful but might cause offence to the more expert then either refuse them or publish as is and add the improvements in a clearly separate way. But, whilst on this theme, I wonder how many of those who would wish to use an alphabetical sort are the least bit interested in having the computer print out the time it had taken to do the job? This addition, to my mind, is a gimmick solely for those whose interest is in techniques rather than the task. I certainly would never use it myself except when attempting to speed up a program by modifying it - another instance where the add-on technique might have been more appropriate than actual incorporation within the original.

One further point on the Beeblet in general - the numbers of misprints and/or spelling and grammatical errors in articles are distracting to pedants like myself. With such elementary mistakes so frequent one begins to wonder just how carefully programs themselves have been checked. Whilst I am only too well aware of how easy it is to slip up in such respects when checking on-screen in word-mode, I suggest that authors could be asked to scrutinise a print-out, or their screen, more carefully. The version I sent you was the third because I usually only spot the mistakes at the instant they are being printed out! But, E & OE, I did check it carefully out of respect for both the magazine and its editor just as I expect contributors to do for me (the occasional paper has been returned by to author(s) for this reason). It did cause me some 'grief' that the PS had more than its fair share of such mistakes when my article carried a University address!

Chris O'Brien:
In the March issue of the magazine, I was disturbed to see a program submitted by J.M.A. Brown, of Auckland, published with alterations made by another magazine contributor. Mr Brown has produced a program which he uses to solve a real life problem. He provided background to the problem, a fairly good description of how the program works, and a comprehensive set of notes on areas which could do with improvement, the rationale behind certain features, and how other hardware would alter program limitations. All in all, a commendable effort which I enjoyed reading, although I have no use for the program myself.

Unfortunately, Steven Cosgrove was apparently given free rein to alter Mr Brown's program (although he was good enough to preserve the 'style'). His comments regarding structured programming, meaningful variable names, documentation, and screen display during operation, were all valid, but he had no right to offer them as a tactless critique of Mr Brown's work, or to arrogantly implement his personal programming standards in work which was not his own.

I do not wish to cast scorn on Mr Cosgrove's obvious enthusiasm for the Beeblet and the considerable effort he puts into contributions - his test programs made good partners for the original program (although assurance that 'it worked' was enough for me), and Electron owners no doubt enjoy his column on the Beeb's little brother. But this aberration in publishing standards must not be allowed to reoccur. If this happens, I cannot imagine any member wishing to contribute and risk the indignities suffered by Mr Brown.

Bruce replies:
ALTERING SUBMISSIONS TO BEEBLET
1) Commments are quite valid - submissions shouldn't be changed without reference back to author

2) Reasons for not consulting simple - at the time more consultation, and therefore more time in preparation = no magazine.

Other consideration : wish to preserve certain design and programming standards which have been published from time to time in Beeblet.

MISTAKES
Yes, these are a problem and do indeed detract from Beeblet. Articles often require extensive upgrading (and usually several draft printouts) before publication. Professor Brown's submission was indeed unusually well prepared. Perhaps someone with time to more thoroughly proof read Beeblet AND make all the necessary amendments would like to assume the editorship?

AF: being the person who typed up the article and listing into a proof-read file for the Editor, I too would like to comment on the alterations, unfavourably:
1. There have to be different techniques in writing programs for different purposes. What I saw was one that was having to consider memory available; single statement lines were inappropriate, amongst other factors.

2. That the bubble sort had gotos in it is perhaps more a sad reflection on the presentation of sorts in Beeblet and other magazines that don't use gotos - if the 'purists' cannot explain the techniques of their style to learning programmers, then they can expect their standards not to be used.
Page end stops my continuing.

PROGRAM SCRAMBLER

Eliot

These two programs are designed to protect a Basic program from being listed or run without the help of the second program. They work rather like the police radio scramblers: the first program scrambles the Basic program as it saves it, while the other loads in and unscrambles the scrambled program. Both programs are stored at the normal PAGE address. This is &1900 for BBC with DFS, &E00 for tape systems, and &1200 for Econets. The Basic program is expected to be stored at PAGE + &100 - that is, &1A00 for DFS, &F00 tape, or &1300 Econet.

To use the Saver program, load it in as a normal program, then type in:
PAGE = PAGE + &100   and Return
Then either load in or type your Basic program. If you press Break, type in
OLD
PAGE = PAGE + &100
to get both your program and the Saver back again.
When you have finished typing or loading your program, type in
PAGE = PAGE - &100
RUN
and the Saver will prompt you from here on.

To load the scrambled program back in,
LOAD the loader program first then type RUN
Follow the prompts. When it has finished loading, type in
PAGE = PAGE + &100
and that will get you back in to your program again.

Note that in this state you will be able to save your program normally.
To make this Saver and Loader unique to yourself, change the values in line 80 of Saver, and line 40 of Loader. DO NOT change the order of the operations on either line.

LOADER:

```
    5 loop=0
   10 P%=PAGE+&100
   15 INPUT"File name",A$
   20 X=OPENUP(A$)
   30 REPEAT
   40   ?(P%+loop)=(BGET#X EOR 10)-6
   45   loop=loop+1
   50   UNTIL EOF#X
   60 CLOSE#X
```

```
SAVER:
¦¦¦10¦*KEY0PAGE=PAGE+&100¦MP%=PAGE¦ML.¦ML%=TOP¦MPAGE=
PAGE-&100¦MPROCsave¦M
    20 *FX 138,0,128
    30 END
    40 DEFPROCsave
    50 INPUT"FILE NAME",A$
    60 X=OPENOUT(A$)
    70 FOR loop=P% TO L%
    80   BPUT#X,?loop+6 EOR 10
    90   NEXT
   100 CLOSE#X
   110 ENDPROC
```

HALL OF FAME

If you can beat the scores in April issue, which haven't changed for several months, let us know. Meantime, here is the latest update:

Felix in Factory	18,000	Felix
Arcadians	42,320	Felix
'747'	745	Felix

WELLINGTON AGM

The minutes of the previous AGM were read by BBC computer using Computer Concept's speech. Beryl Dixon and Anne Flower have resigned from the committee, and been replaced by:
John Bocock, John Colvin, and Neville Hudson will liaise with the National committee.

NEXT WORKSHOPS:
May 28th at Wellington College
June 25th venue ditto.
Workshops are consistently the 4th Wednesday of each month.

MONTHLY MEETINGS:
At the Royal Society rooms.
Dates for the rest of the year:
June 18th (Weds) - Masters and Modems
July 16th (Weds)
August 13th (Weds)
September 10th (Weds)
October 9th (Thurs)
November 12th (Weds)

USER GROUP MEETINGS

AUCKLAND: -meets 2nd Wednesday of the month, 7.30pm, Conference Rooms, Auckland Secondary Teachers College, 60 Epsom Ave, Epsom. ph Dave 770-630 x 518(bus) or Kerry 695-355.
-North Shore: meets last Wednesday of the month, 7.30pm, Salvation Army Hall, Glenfield Road, Glenfield ph Chris 419-0543 (wk)
CHRISTCHURCH: -fortnightly workshops, Monday 6.30pm at Hagley High School. ph. Michael 582-267.
DUNEDIN: -first Thursday of the month. Phone Martin McDowall 42-031
HAWKES BAY: -Alternative months Hastings & Napier. Contact Kendall Napier 435-624, Bob Taradale 446-955, Mitch Hastings 778-235.
INVERCARGILL: -Phone Evan 330-444 Invercargill.
PALMERSTON NORTH: -Phone Tom Skinner 83-009(wk)
TAUPO: -meets every second Wednesday @ 7pm. Contact David 84-215 (h), 85-124 (wk).
TAURANGA: -every second Wednesday. ph. Chris & Jane 65-076.
TIMARU: -Phone Lloyd van der Krogt Timaru 61-412.
TOKOROA: -meets first Friday of each month, Tokoroa High School, 8pm. Phone Brian 67-625 Tokoroa.
WAIKATO: -first Friday of the month 7.30pm Waikato Tech Institute. Ph. Alison or John, Morrinsville 6695.
WELLINGTON: -meets usually second WEDNESDAY of the month, 7.30pm, Royal Society Lecture Theatre, Turnbull St, Thorndon. Other dates advised in Beeblet regional news.
-Workshops: fourth Wednesday of month, other venues. ph. Don 848-235 or Mike 785-437.
OTHER CENTRES: - let us know and we will publish details here.

THE NEWSLETTER:

CONTRIBUTIONS:-most welcome. Listings should be sent in on tape or disc. (Tape - 2 copies, one at 300 baud please). Please include written explanation of listing; text and graphics only material welcome too.

DEADLINES:- For a particular issue, the last day for material is the last day of the month prior to publication.

ADVERTISING:- Rates are $20 per half page (camera ready), deadline as above.

Copyright (c) BBC/ACORN COMPUTER USER GROUP OF NZ INC.
Published monthly except January and mailed to financial members.

MEMBERSHIP:

Membership of the users group is on payment of an annual subscription from April to March. For the 1986/7 year it is: $30.00, plus $5.00 for Wellington members as a local branch levy, which entitles attendance at meetings as detailed above. Joining or renewing members get all the back issues of the current year - those joining from 1 October may have half the annual sub waived and only receive newsletters from 1 October.

THE BBC/ACORN COMPUTER USER GROUP OF NZ INC, P.O.Box 9592, WELLINGTON.

Reproduced by BRYCE FRANCIS LTD ,WELLINGTON